

Computer Practical: Clouds and Orographic Precipitation

Paul Connolly and Ann Webb, May 2020

1 Overview

In this computer practical, a single column model of the atmosphere (SCM) is used to demonstrate a number of phenomena, including: the importance of aerosol-cloud interactions; the importance of the warm-rain process; and the importance of desert dust as ice nucleating particles (INP). This model solves equations for how aerosol and cloud particles evolve into precipitation particles. These equations are used in many weather forecasting models.

2 Model formulation

There are many equations in this model and I cannot write them all down here; however, we can gain an appreciation of how the the ODEs and PDEs that the model solves with the Equations below.

$$\theta = T \left(\frac{1 \times 10^5}{P} \right)^{0.286} \quad (1)$$

$$T = \theta \left(\frac{P}{1 \times 10^5} \right)^{0.286} + \frac{L_v}{c_p} \frac{dm_i}{dt} \Big|_{vap} \quad (2)$$

$$\frac{dm_i}{dt} \cong \frac{a\pi D_i^{2+b}}{4} \sum_j \frac{\rho_w \pi}{6} d_j^3 n_j + \frac{dm_i}{dt} \Big|_{vap} \quad (3)$$

$$\frac{dn_i}{dt} = -n_i \sum_j \frac{a\pi D_j}{4} \quad (4)$$

$$\frac{\partial \theta}{\partial t} + w \frac{\partial \theta}{\partial z} = 0 \quad (5)$$

$$\frac{\partial q_v}{\partial t} + w \frac{\partial q_v}{\partial z} = 0 \quad (6)$$

$$\frac{\partial n_i}{\partial t} + (w - v_{i,fall}) \frac{\partial n_i}{\partial z} = 0 \quad (7)$$

Here, θ is potential temperature; T temperature; P pressure; L_v is the latent heat of vapourisation; c_p is the heat capacity of the air; $\frac{dm_i}{dt} \Big|_{vap}$ is the growth of drops from water vapour; a and b are both constants describing the fall-speed of drops; D_i is the diameter of large drops collecting small drops of diameter d_j with a number concentration n_j ; ρ_w is the density of liquid water; w is the vertical wind-speed; q_v is the water vapour mixing ratio; and $v_{i,fall}$ is the fall-speed of drops.

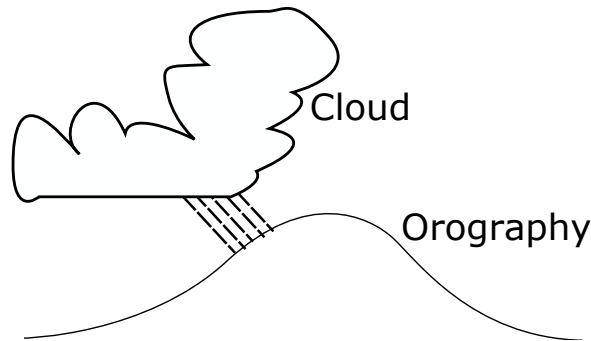


Figure 1: Schematic of the scenario being modelled, where the orography is the mountain or hill, and clouds become deeper and precipitate as they are lifted over the orography.

The one-dimensional model is written in Fortran (like many other models). The model reads an input file called `namelist.in`. When the model is run it generates an output file, which is a format known as NetCDF. NetCDF is a common data format used in atmospheric and environmental sciences. Data from the NetCDF file can be read by Python script to plot out the results.

The model has *prognostic* variables for aerosol, cloud, rain, ice, and temperature. Clouds are represented by the values of these variables as a function of height and time. When the cloud is shallow, it tends to not precipitate and stays in the atmosphere indefinitely. However, when the cloud deepens water condenses on the cloud drops to make them large enough to rain. Figure 1 illustrates the main idea behind the model simulations.

3 Downloading and compiling the code

Log into the server computer with an SSH window. Also, log into the server computer with an SFTP window. Space the windows out on your screen so that you can easily move between them.

If you have not done this already, download the code you will be using today from GitHub by typing the following:

```
git clone ssh://git@github.com/EnvModelling/simple-cloud-model
```

this should download the code to your working directory.

In order to access the files you will need to change the working directory to where the files are. Type the following:

```
cd simple-
```

followed by the `tab` key and the command should auto-complete. Then press the `enter` key.

Type `ls` followed by the `enter` key. The screen should list the files in this directory.

Finally, type `make` followed by the `enter` key. This will compile the FORTRAN code into machine code that the computer can run.

4 Viewing and Editing the code

We will use the `nano` text editor to view some files. In this practical there are two input files: one is called `prac/namelist_prac.in`. To see what variables we can change, type:

```
nano -l prac/namelist_prac.in
```

the `-l` means to show line numbers. The code can be edited in the text editor and saved by typing `Ctrl-X` at the same time and pressing `Y` to save the file. The important lines are lines 24 and 39-44 of this file.

On lines 24 the initial cloud drop number is defined.

On lines 39-44 the variables `tsurf`, `t_cbase`, and `t_ctop` are defined.

We input the aerosol in a separate file called `prac/namelist.pamm.bam.in`. To edit this file, type:

```
nano -l prac/namelist.pamm.bam.in
```

The important lines are lines 12-13 of this file, where the number of aerosol particles in each of 3 modes, `n_aer1`, are defined.

5 Running the model

After the code has been compiled (see Section 3), the procedure for running the model is to type:

1. Edit the files `namelist_prac.in` and `namelist.pamm.bam.in` to configure the model.
2. Type `./run.sh` at the command line, followed by `enter` to run the model.

Table 1: Parameters controlling the behaviour of the model.

Variable	Default value	Description
N_{cloud}	100 cm ⁻³	Initial number conc. of cloud drops <code>prac/namelist_prac.in</code> , line 24.
N_{dust}	0.1 cm ⁻³	Number conc. of desert dust particles, <code>prac/namelist.pamm.bam.in</code> , line 12-13
ΔT	2 K	Temperature of cloud base minus temperature of cloud top. These are both set on lines 41-42 of <code>prac/namelist_prac.in</code>
$T(K)$	303.15 K	Atmospheric temperature at sea-level, <code>prac/namelist_prac.in</code> , line 39.
<i>Warm rain</i>	<code>.true.</code>	Whether the simulation includes the warm-rain process or not, <code>prac/namelist_prac.in</code> , line 20.

Table 1 describes the parameters that can be modified. Note that a typical cloud drop concentration in the clouds is 100 cm⁻³. This means if you count the number of cloud drops in 1 cm³ there would be 100 of them!

In the atmosphere cloud drop number concentrations are linked to aerosol particle concentrations since cloud droplets form around these aerosol particles. Generally, aerosol concentrations are higher in more polluted environments. A typical cloud drop concentration in a polluted environment might be 2000 cm⁻³.

After running the model a NetCDF file will be generated at the location `/tmp/<username>/output.nc`. You can then run a python script by typing:

```
python3 python/scm_plot.py
```

and a plot of the model simulation will be shown in `/tmp/<username>/scm_plot.png`, which you can download using SFTP by typing:

```
get /tmp/<username>/scm_plot.png
```

6 Experiments

FOR ALL EXPERIMENTS: I STRONGLY SUGGEST YOU OPEN UP A WORD OR POWERPOINT DOCUMENT AND INSERT THE FIGURES AND MAKE SOME NOTES AS YOU GO. YOU COULD BE ASKED QUESTIONS ABOUT THEM IN THE ASSESSMENT.

6.1 Changing Cloud Droplet Number

Run the model for the default case by typing `./run.sh` from the `simple-cloud-model` directory. Then plot the output by typing `python3 python/scm_plot.py`.

After a short while Python will put a file in the `ls /tmp/<username>/` directory. To download it, go to your SFTP window and type:

```
get /tmp/<username>/scm_plot.png
```

followed by `enter`, where `<username>` is your username. This plot will be discussed in the practical.

Look at plot (a). You will notice that the cloud disappears as it rises over the mountain (drawn in red). Why is this?

Look at the rain in plot (c). You will notice that as the cloud disappears the rain appears and falls to the surface. This is because cloud droplets are becoming large enough to collide and coalesce to form rain drops, which is known as the *warm-rain process*.

Repeat the simulations for:

- $N_{cloud} = 10 /cc$ ($10 \cdot e6$, i.e. lower than normal)
- $N_{cloud} = 1000 /cc$ ($1000 \cdot e6$, i.e. higher than normal)
- $N_{cloud} = 2000 /cc$ ($2000 \cdot e6$, i.e. very polluted)

Do not forget to save the images each time.

Question: For the default case where on the mountain does the rain fall? What do you notice as you reduce, and increase the cloud drop number concentration (in terms of where the rain falls)?

6.2 Switching off the Warm Rain Process

In this simulation change all values back to their defaults in Table 1 and then turn off the warm rain process by setting it to `.false.` on line 20 of `prac/namelist_prac.in` and run the model. Repeat the simulation with `warm_rain` switched off and the number of cloud drops set to $2000 /cc$ ($2000 \cdot e6$). Create your plots and make some notes.

Question: What do you notice about the persistence of the clouds and the rain fall when switching off the warm rain process?

6.3 Cooler surface conditions

Again change all values back to their defaults in Table 1, then set the surface temperature to 280 K, the cloud-base temperature to 270 K and the cloud-top temperature to 268 K. Run the model and create the plot. Repeat the simulation with `warm_rain` switched off. Looking at plot (d) you should notice that there is now ice in the simulations (N_{ice} is the number of ice crystals in 1 litre of air).

Question: What do you notice in the simulation without warm rain? Is there any rain fall? How can this be?

For the final simulation keep the *warm rain process* turned off and increase the value of N_{dust} to $1000 /cc$.

Question: How does increasing the number of dust particles affect the number concentration of ice crystals in the simulations?

6.4 Deeper Cloud

Finally, change all values back to their defaults in Table 1 again. Now change the value of N_{cloud} to $1000 /cc$ and ΔT to 5 degrees (i.e. cloud-base temperature set to 293.15 K and cloud-top temperature to 288.15 K). Run the model and generate the plot. You will see that the cloud is around 1 km deep now.

Exercise: Compare the two simulations with $N_{cloud} = 1000 /cc$ and $\Delta T = 2$ or 5 degrees. How does the value of ΔT affect the amount of precipitation and why?